

New Database Manipulation Tools in the Easy-Learning On-Line Platform

Radu Rădescu
Polytechnic University of Bucharest,
Bucharest, Romania

Andrei Davidescu
Credit Express Financial Services SRL,
Bucharest, Romania

Valentin Pupezescu
Polytechnic University of Bucharest,
Bucharest, Romania

The present paper deals with the new ORM (object-relational mapping) tool introduced in the easy-learning platform. Propel 1.5 is the latest version of Propel, one of the ORMs fully compatible with the Symfony framework, and in comparison with the older versions and it has drastically improved the way the easy-learning platform can manipulate its database. Being a complex platform, the database behind it is complex as well and can contain a large number of entries that can increase the time required to load pages, slowing down the whole application. Propel 1.5 provides queries which are minimized and optimized, written and executed faster and the speed of retrieving information from the database is also improved. Besides these aspects, Propel 1.5 comes with a new behavior that provides a robust database, reducing the risk of losing references among tables, allowing the administrators of the easy-learning platform to temporarily delete objects and giving the possibility of restoring them in case they are needed once more. Another feature is the admin15 theme provided to the Symfony admin-generator files, making them easy to manage, edit and improve. Using this tool has three purposes: speeding up the platform, providing a robust database and giving the platform the ability to be maintained more easily.

Keywords: e-learning, on-line educational platform, database programming

Introduction: ORM (Object-Relational Mapping) Tool

An ORM is a programming technique used at converting data among incompatible type systems in object-oriented programming languages. The result is a “virtual object database” that can be used from within the programming language. Many popular database products, such as structured query language database management systems can only store, and manipulate scalar values, such as integers and strings organized within normalized tables. These object values can either be converted into groups of simpler values for storage in the database or only use simple scalar values within the program.

ORM can be used to implement the first approach. The main problem is translating those objects to forms that can be stored in the database for easy retrieval, while preserving the properties of the objects and their relationships; these objects are then said to be persistent.

Databases are relational. PHP (hypertext preprocessor) and Symfony are object-oriented. In order to access the database in an object-oriented way, an interface translating the object logic to the relational logic is required. This interface is called an ORM, which is made up of objects that give access to data and keep

Radu Rădescu, Department of Applied Electronics and Information Engineering, Polytechnic University of Bucharest.

Andrei Davidescu, Department of IT Administration, Credit Express Financial Services SRL.

Valentin Pupezescu, Department of Applied Electronics and Information Engineering, Polytechnic University of Bucharest.

business rules within themselves. One benefit of an object/relational abstraction layer is that it prevents from using a syntax that is specific to a given database. It automatically translates calls to the model objects to SQL (structured query language) queries optimized for the current database.

An abstraction layer encapsulates the data logic. The rest of the application does not need to know about the SQL queries, and the SQL that accesses the database is easy to find. Using objects instead of records and classes instead of tables has another benefit: there can be added new accessory to tables.

For instance, if there is a table called “student” with two fields, first name and last name, it might be needed to retrieve just a “name”. In an object-oriented world, this is as easy as adding a new accessory method to the “student” class, like that in Figure 1.

```
public function getName()
{
    return $this->getFirstName().' '.$this->getLastName();
}
```

Figure 1. New accessory method to the “student” class.

Easy-Learning, Symfony 1.4 and Propel 1.5

The latest version of the easy-learning platform is developed with the aid of Symfony 1.4, one of the most powerful open-source PHP5 frameworks. Symfony is a full-stack MVC (model-view-controller) framework that helps at developing Websites faster. It also establishes a set of best practices that will help to develop maintainable and secure Websites.

Symfony is a complete framework designed to optimize the development of Web applications by way of several key features. For starters, it separates a Web application’s business rules, server logic and presentation views. It contains numerous tools and classes aimed at shortening the development time of a complex Web application. Additionally, it automates common tasks so that the developer can focus entirely on the specifics of an application. The end result of these advantages means that there is no need to reinvent the wheel every time a new Web application is built!

Symfony is written entirely in PHP. It has been thoroughly tested in various real-world projects and is actually in use for high-demand e-business Websites. It is compatible with most of the available databases engines, including MySQL, PostgreSQL, Oracle and Microsoft SQL Server.

This version of the framework is significantly better than the 1.0 version of Symfony used in the former easy-learning platform. It provides a new default mailer based on SwiftMailer 4.1, new widgets, validations, a new and versatile form sub-framework, autoloaders, support for more powerful plug-ins, and last but not least, support for the new Propel 1.5 ORM. Propel 1.5 is an open-source ORM for PHP5. It allows access to the database using a set of objects, providing a simple API (application programming interface) for storing and retrieving data. It uses PDO (PHP data objects) as an abstraction layer and codes generation to remove the burden of runtime introspection, therefore, it is a fast ORM.

Propel 1.5 implements all the key concepts of mature ORM layers: the active record pattern, validations, behaviors, table inheritance, reverse engineering an existing database, nested sets, nested transactions, lazy loading and LOB (large objects).

The relationship of Symfony and Propel is shown in Figure 2.

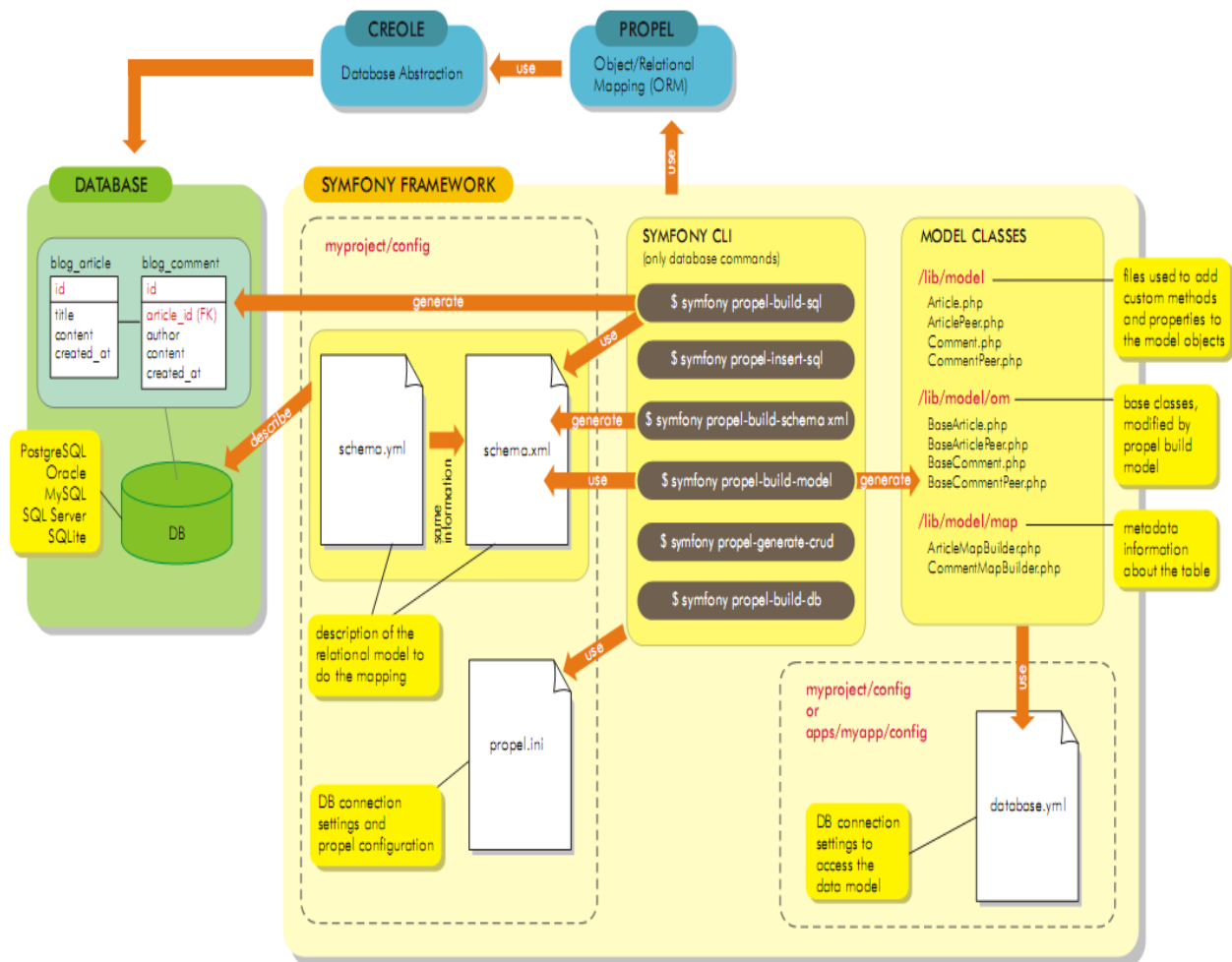


Figure 2. The relationship of Symfony and Propel.

Query API (Application Programming Interface)

Along model and peer classes, Propel 1.5 generates one query class for each table of the easy-learning database. These query classes inherit from criteria, but have additional abilities, since the propel generator has a deep knowledge of the defined schema. That means that Propel 1.5 recommends the usage of model queries instead of raw criteria.

Model queries have smart filter methods for each column and termination methods on their own. That means that the following code (see Figure 3).

```
<?php
$c = new Criteria();
$c->add(StudentPeer::NUME, 'Ionescu');
$student = StudentPeer::doSelectOne($c);
```

Figure 3. Former student-model-query.

It can be replaced by the one in Figure 4.

```
<?php
$q = new StudentQuery();
$q->filterByNume('Ionescu');
$student = $q->findOne();
```

Figure 4. New student model query.

In addition, each model query class benefits from a factory method called “create”, which returns a new instance of the query class. In addition, the filter methods return the current query object. Therefore, it is even easier to write the previous query in Figure 5.

```
<?php
$book = StudentQuery::create()
->filterByNume('Ionescu');
->findOne();
```

Figure 5. Short student model query.

These commands are transformed into minimized and optimized database queries, which are executed faster, resulting in a faster information retrieval. This is a very important aspect in the easy-learning platform considering the large number of entries in the database and that the users, administrators, teachers or students need specific information forming it. This unique Propel 1.5 feature makes the easy-learning platform faster and easier to use, administer and manage.

Propel 1.5 eases the way of finding objects related to a model object that is already know. The developer has the advantage of the generated filter by “XXX” methods in the query objects, where “XXX” is a relation name (see Figure 6).

```
<?php
$serie = SerieQuery::create()->findPk(1);
$grupe = GrupaQuery::create()
->filterBySerie($serie)
->orderByNume()
->find();
```

Figure 6. Retrieving related objects.

There is no need to specify that the “serie_id” column of the “Grupa” object should match the id column of the “Serie” object. Since it has been already defined the foreign key mapping in the schema, Propel knows enough to figure it out.

The “Soft_Delete” Behavior

Behaviors are a great way to package model extensions for reusability. They are the powerful, versatile, fast, and help at organizing the code in a better way. The “soft_delete” behavior overrides the deletion methods of a model object to make them hide the deleted rows but keep them in the database. Deleted objects still do not show up on select queries, but they can be retrieved or undeleted when necessary.

This feature is used on specific tables from the database, mainly the tables that provide foreign keys to other objects. In the easy-learning platform, the student, teacher, course, test and poll tables have “soft_delete”

behaviors implemented. For each of these tables, propel automatically creates a “deleted_at” column having by default a NULL (zero) value and automatically updates the query classes.

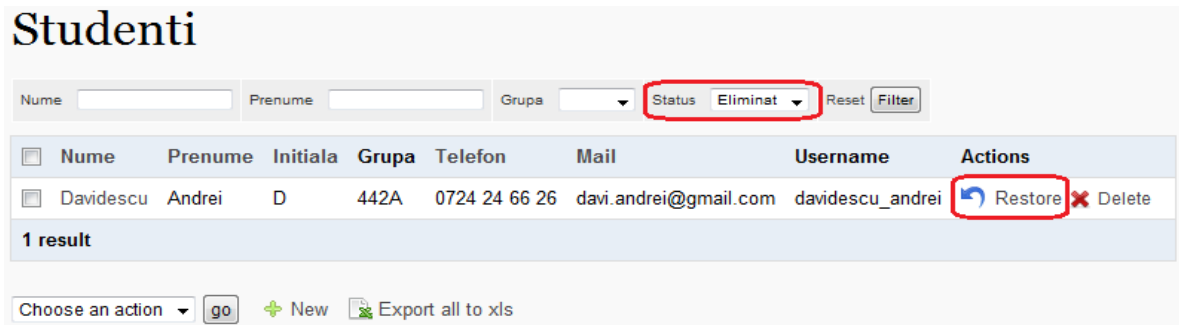


Figure 7. Restoring a deleted student.

As seen in Figure 7, when an object of that type is deleted, its “deleted_at” property is set with the date and time of the action. The query classes are set to ignore the objects, which do not have the “deleted_at” column NULL. In this way, even though these object exist in the database, they are hidden so that they appear as deleted and can be revealed at any time by using the undelete method.

In the previous version of the easy-learning platform, when an important object, such as a student which has information in related tables, for example grades, was deleted, the reference to those grades were lost, resulting in a bad organized database. The “soft_delete” behavior allows the administrator and the teacher to restore the objects deleted by mistake.

Admin 15 Generator Theme

Propel 1.5 comes bundled with a new admin generator theme named “admin 15”. This theme provides additional features to the easy-learning platform, based on the new Propel 1.5 query objects, and is backwards compatible with “sfPropelPlugin’s” admin generator theme.

The admin15 theme does not use the peer classes anymore; therefore, settings referencing the peer classes are ignored in this theme. This includes “peer_method” and “peer_count_method”. The new theme provides a simple alternative for these settings, called “with”. There can be added each of the objects to hydrate together with the main object in the “with” setting list. Hydrating the objects is a faster way to retrieve large amounts of information from related tables in the database in only one query.

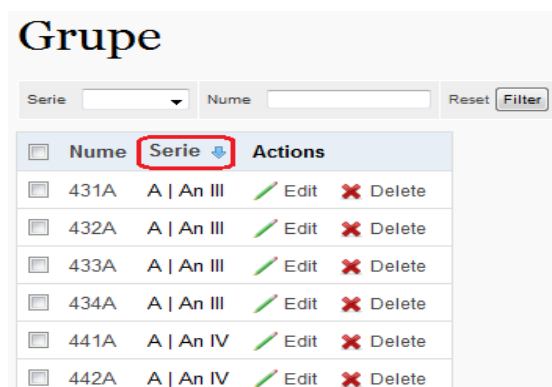


Figure 8. Sortable related series column in the groups list view.

As shown in Figure 8, the new theme provides an easy way to make virtual columns and foreign key columns sortable in the list view. Just declare the corresponding fields which is sortable to true, and the generated module will look for an order by “XXX” method in the generated query. For instance, it allows a groups list to be sortable on the series name.

Conclusions

The new Propel 1.5 is a very important tool for the new version of the easy-learning platform. The ORM speeds up the platform by using the new model query, provides a more robust database by implementing the “soft_delete” behavior and makes the platform more user-friendly by using the new admin15 generator theme.

References

- Bååth, J. A. (1982). *Distance students' learning—Empirical findings and theoretical deliberations* (pp. 30-32). Stockholm.
- Nagy, A. (2005). E-content: Technologies and perspectives for the European market. *The impact of e-learning* (pp. 79-96). Berlin, Springer Verlag.
- Rădescu, R. (2004). E-learning: Concepts, implementation and applications, *IT&C Market Watch, Fin Watch*, 50 (No. 30/2004), 61 (No. 31/2004), 50 (No. 33/2004), Bucharest.
- Rădescu, R., & Iovan, R. (2004). Improvements to the easy-learning e-learning platform (pp. 275-278). In proceedings of the *Fifth European Conference E-COMM-LINE*, Bucharest.
- Rădescu, R., & Mărescu, R. (2005). External use of the easy-learning platform: A Web-based application. In proceedings of the *Symposium “Educational Technologies on Electronic Platforms in Engineering Higher Education” (TEPE 2005)* (pp. 227-234). Technical University of Civil Engineering of Bucharest, May 27-28, 2005, Bucharest.
- Rădescu, R., & Iovan, R. (2005a). Generating the class register in the easy-learning platform. In proceedings of the *National Conference of Virtual Education “Virtual Learning—Virtual Reality”*. *Educational software and management* (pp. 213-220) (3rd ed.). University of Bucharest, Mathematics and Informatics Faculty, Bucharest.
- Rădescu, R., & Iovan, R. (2005b). Creating and using tests in the easy-learning platform. In proceedings of the *National Conference of Virtual Education “Virtual Learning—Virtual Reality”*. *Educational software and management* (pp. 229-235) (3rd ed.). University of Bucharest, Mathematics and Informatics Faculty, Bucharest.
- Rădescu, R., & Iovan, R. (2005c). New facilities of the easy-learning platform. In proceedings of the *Symposium “Educational Technologies on Electronic Platforms in Engineering Higher Education” (TEPE 2005)* (pp. 219-226). Technical University of Civil Engineering of Bucharest, May 27-28, 2005, Bucharest.
- Rădescu, R., & Bojin, M. (2006). Function generators in the easy-learning platform. In proceedings of the *National Conference of Virtual Education “Virtual Learning—Virtual Reality”*. *Educational software and management* (pp. 115-120) (4th ed.). University of Bucharest, Mathematics and Informatics Faculty, Bucharest.
- Rădescu, R. (2007a). Test user interface in the easy-learning platform. In proceedings of the *National Conference of Virtual Education “Virtual Learning—Virtual Reality”*. *Modern methods in education and research* (pp. 85-92) (5th ed.). University of Bucharest and “Ovidius” University of Constanța, October 26-28, 2007, Bucharest.
- Rădescu, R. (2007b). Test management interface in the easy-learning platform. In proceedings of the *National Conference of Virtual Education “Virtual Learning—Virtual Reality”*. *Modern methods in education and research* (pp. 75-84) (5th ed.). University of Bucharest and “Ovidius” University of Constanța, October 26-28, 2007, Bucharest.
- Rădescu, R., & Urse, C. (2007a). Graphic tools in the easy-learning platform. In proceedings of the *8th European Conference E-COMM-LINE*. SIV-26e/1...6, Bucharest.
- Rădescu, R., & Urse, C. (2007b). Advanced testing methods in the easy-learning platform. In proceedings of the *Eighth European Conference E-COMM-LINE*. SIV-26e/1...6, Bucharest.
- Rădescu, R. (2008a). Class register's optimization in the easy-learning platform. In proceedings of the *National Conference of Virtual Education “Virtual Learning—Virtual Reality”*. *Modern methods in education and research* (6th ed.). University of Bucharest and “Ovidius” University of Constanța, October 31-November 2, 2008, Bucharest, B-7-55/1...4.

- Rădescu, R. (2008b). Multiple tests in the easy-learning platform. In proceedings of *the National Conference of Virtual Education "Virtual Learning—Virtual Reality"*. Modern methods in education and research (6th ed.). University of Bucharest and "Ovidius" University of Constanța, October 31-November 2, 2008, Bucharest, B-6-54/1...4.
- Scott, W. A. (2000). *Mapping objects to relational databases: O/R mapping in detail, practice leader, agile development*. Developer Works.
- http://www.symfony-project.org/book/1_0/
- <http://www.zend.com/zend/zend-engine-summary.php>
- http://forge.mysql.com/wiki/MySQL_Internals
- <http://atm.neuro.pub.ro/radescu>
- <http://easy-learning.neuro.pub.ro>